Appendix J:
Technical Architecture

DIT Vision

The formation of our vision started by listening to our stakeholders. Michigan's IT Strategic Plan was developed after an extensive research process to determine citizen and government priorities. The plan also reflects mandates and information collected from relevant executive orders, recently conducted town halls, established advisory groups, and multiple meetings with staff.

"A connected Michigan where access is just a click away, where services are streamlined and secure, and where citizens have an immediate voice in an open and energetic public square."

DIT's vision, goals, objectives, and strategies were developed after extensive meetings to determine what our customers and employees believe are the priorities for state information technology. It also reflects information collected from Executive Orders, audit reports, town halls, and informal meetings with operations staff.

Michigan's technology vision is the center of the technical architecture. Technology decisions start with a business need or imperative. The vision gives us the goal our architecture must obtain to create a "Connected Michigan."

Purpose

This document serves to outline the technology elements that collectively make up the components of Michigan's Technical Architecture. It will be used to:

1. Set the direction for technology decisions during the current strategic planning cycle according to technical architecture guiding principles.
2. Outline the process for creating a comprehensive and ongoing technical architecture, including the development of standards.
3. Define a common vocabulary for technology components that will foster inter-departmental discovery, collaboration and interoperability.

The technical architecture will support the adoption and implementation of product standards and best practices on a statewide scale. Once fully implemented the state will realized enormous benefits and economies of scale and align capital investments to the enterprise solutions and standards being developed.

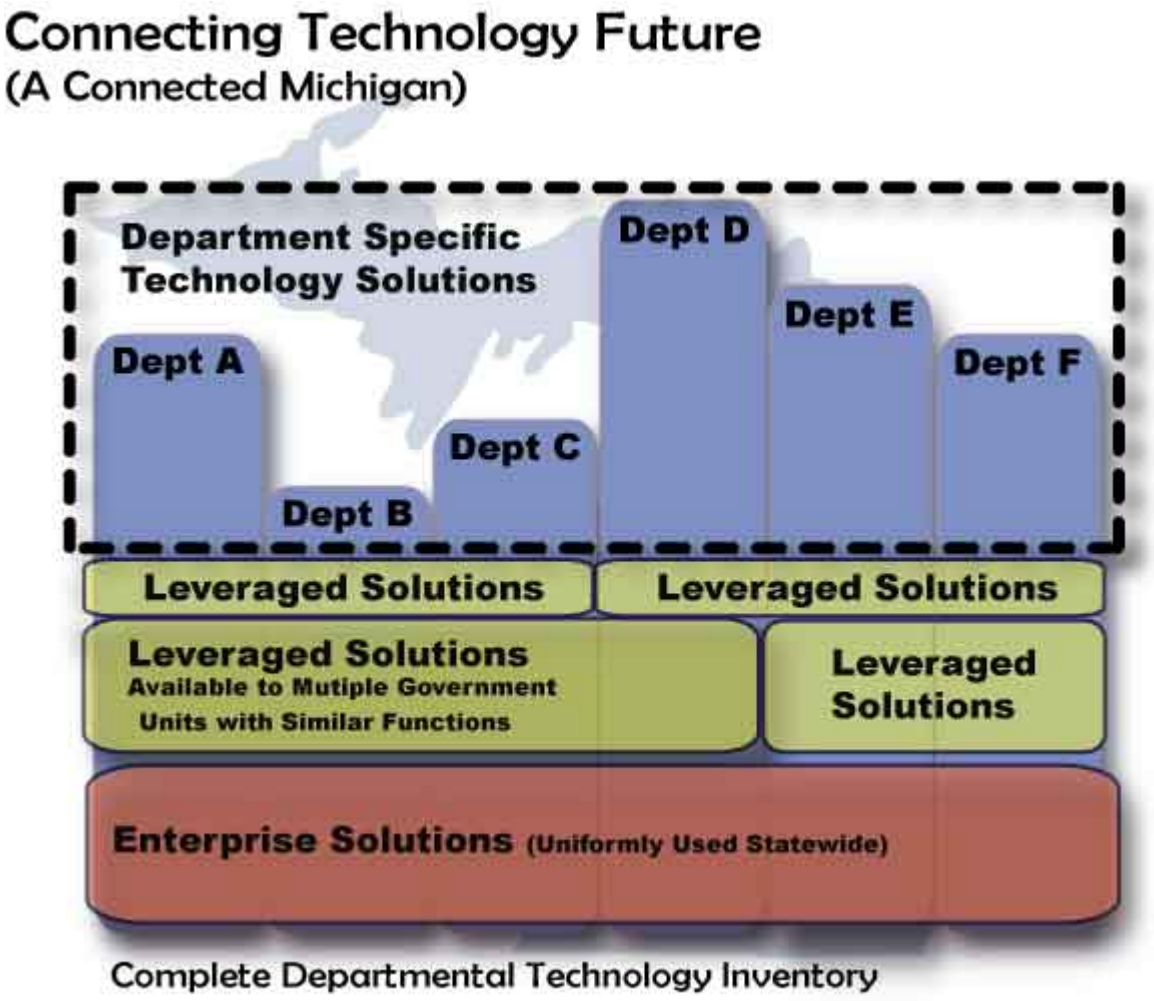


Figure 2: Federated Technology Leveraging Strategy

The technique being employed by DIT is a federated approach. A federated approach recognizes that leveraging can occur on a statewide basis or on a more limited basis surrounding a function or area of common need affecting only a few departments. There will always be the need to recognize the technological autonomy of any given agency to resolve a business issue. This autonomy, however, must always be balanced with the greater good created by sharing information and technology solutions statewide. DIT's internal processes and governance model create an objective framework for us to transform our collection of individual departments and their technological capabilities into a

"whole state" approach by accounting for the many competing priorities to be addressed and ensuring that DIT meets and supports the business needs of the state in the decisions it makes on information technology resources.

The creation of the Michigan Department of Information Technology (DIT) has enabled the State of Michigan to consolidate both software and hardware technologies where appropriate. To do that, it was imperative to commission the development of Enterprise Level Architecture. Numerous independent studies conducted prior to the formation of DIT recommended that although much had been accomplished in the area of a statewide architecture, a number of challenges / opportunities still exist to provide cost savings and efficiencies for Michigan. The Department of Information Technology is committed to expanding, improving and enforcing the Technical Architecture throughout the state to become a leader in leveraging solutions across departments and expanding standard practices for all technical solutions.

Credits and Disclaimers

Appreciation for the work accomplished and acknowledgement for the framework and portions of the information must be credited to The Technical Reference Model (TRM) Version 1.1 produced by the Federal Enterprise Architecture Program Management Office (FEAPMO) and the enterprise architecture framework developed by the National Association of State CIO's (NASCIO). Many of the terms and their definitions were taken directly from these sources; other information was inserted or changed to reflect the design and requirements for the State of Michigan.

This iteration of the technical architecture is not intended to endorse a particular vendor product.

Setting the Direction: the Technology Focus for 2003-2007

The Technical Architecture Model concentrates the limited technology planning and design resources within DIT. This focus is derived first from business goals and initiatives outlined in the three-year Strategic Planning horizon. The Department of Information Technology has outlined its goals for the 2004-2007 within the Strategic Plan.

Just as DIT is operationally driven by guiding principles in how it will accomplish its goals and objectives, the architecture itself must adhere to these principles as well. Each of the Guiding Principles derived from our clients and stakeholders has an impact on the Technical Architecture of Michigan. Indeed, many of the principles themselves are architectural guidelines elevated to the level of a DIT Guiding Principle by virtue of their importance to our clients (e.g. "consider COTS (commercial off-the-shelf) whenever possible," "maintain separate development, testing and production regions," etc.).

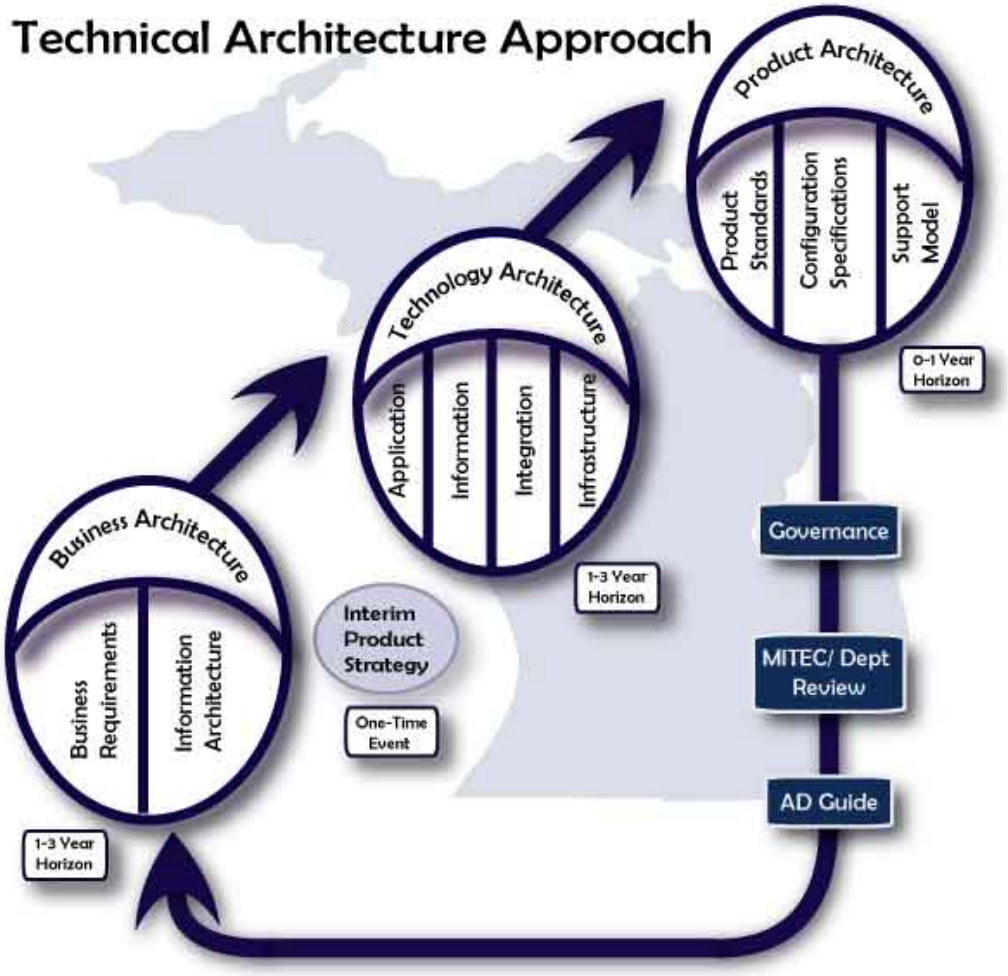
The following list outlines the Technical Architecture Guiding principles derived from the Business (MITEC) Guiding Principles.

Business (MITEC) Guiding Principle	Technical Guiding Principle or Business Requirement (s)
Take The Enterprise View	Leverage within state agencies Leverage across all government entities Establish interoperability standards
We Are In This Together	DIT will engage early to understand the business requirements of a given project or initiative and solve the problem with common and open technology solutions that build agility and maintainability. Business and IT will coordinate technology through a common vision. Support and management of technology resources will be cost effective.
Plan First, Build Once, Use Often	Build and deploy web-enabled applications Build solutions that are efficient and supportable Deploy solutions only when sound technical reviews and testing has taken place. Business requirements and processes must produce an Technical Architecture that is extensible, scalable and adaptive. Provide interoperability while reducing the complexity of integration. Solutions will be built to leverage across multiple departments and governmental entities. Take advantage of existing functions and components that are repeatable and reusable. Build a secure, extensible and open database system that enables the sharing of data elements.
Measure Success	**Limited Technical Architecture Implications*
Secure our Systems	Build Security and Audit into Every Initiative

Business (MITEC) Guiding Principle	Technical Guiding Principle or Business Requirement (s)
	Technical Architecture must include business continuity, security and disaster recovery capabilities. The architecture must provide a system design that includes redundancy that will ensure system stability.
Sponsor Every Project	**Limited Technical Architecture Implications*
Maintain Separate Development, Testing and Production Regions	The configuration specification standards for application development tools must detail environments that support version control and configuration management capabilities.
Consider COTS Whenever Possible	"Commercial-off-the-shelf" solutions will be targeted with appropriate support and maintenance models. We will balance the need for standard technologies statewide with the obvious benefits of COTS solutions.
Life Cycle Strategies	Proactively plan for the eventual migration and elimination of targeted technologies. Consider industry trends, best practices and market forces when designing or updating the technology architecture.
Standards and Architecture Driven	Execute an inclusive, ongoing and decisive technical architecture process. Select proven technologies that provide standard hardware, software configurations and data presentation protocols.

The Process for Change: Technical Architecture Approach

The Technical Architecture Approach outlined below depicts three main steps in the process. Each step is designed to provide the basis that ties all technical decision making back to client and stakeholder business needs. For 2004 the process includes a one-time "Interim Product Strategy" that will serve as a short-term procurement guideline for department technology purchases. This step was deemed necessary after recent analysis discovered that not only are a large number of disparate legacy technologies still being maintained, but also that our complexity is continuing to expand in certain areas through the introduction of "bleeding edge" technologies in new initiatives.



Technical Architecture Approach

Each step in the Technical Architecture Approach is described below:

Business Architecture

This step involves identifying and documenting the needs of our clients and stakeholders. The two main thrusts of the Business Architecture are high-level business requirements and comprehensive information architecture.

Technology Architecture

The Technology Architecture categorizes the technical environment, defines functional standards and outlines the basis for the selection of solutions and specific products. Michigan has developed a clear topology for presenting the Technology Architecture in accordance with the NASCIO and FEAPMO models. Each level of the topology narrows the focus to specific elements of the technical architecture.

A. **Domain** – A natural technology division forming the main building block of the Technology Architecture; each domain may have one or more disciplines
B. **Disciplines** – a logical, functional subset of the domain which is a cohesive unit with regard to its subject areas and stakeholders; each discipline will contain information for one or more technology areas
C. **Technology Areas** – technical topics that support the disciplines containing one or more specific technology categories
D. **Specific Technology Categories** – More detailed and specialized components of the technology areas containing one or more products and solutions (leading to the final selection of a particular solution, product or vendor and outlined in detail in the product architecture).

Product Architecture

The Product Architecture takes the functional requirements outlined in the Technology Architecture and defines selected products and the standards, which govern their implementation and usage. This area includes product selections and standards, engineering/configuration specifications and the support model required for a given solution.

*The technology architecture is the focus of this document, and is discussed in detail later

Deciding on Solutions: Standards Development Process

Standards and their enforcement are the backbone of Michigan's approach to meeting many of its strategic goals and objectives. As such this process plays a major role in the state's technical architecture.

Standards are defined and documented at several levels throughout the Technical Architecture process. There are two chief types of standards being discussed within this process:

Functional Standards

Standards concerned with the overall requirements of a given technology domain or process. These standards define what a technology should accomplish, its integration requirements, environmental limitations and business issues it must resolve.

Product Standards

Standards concerned with specific technology product selections. Including "preferred" versions numbers, engineering and configuration specifications and support model definitions.

The standards process was created to maintain consistency from the initial recognition of a business need to the ultimate selection of technical solution and vendor. For this reason the Department of Information Technology's standards development model overlaps areas within technical architecture and acts a consistent oversight "check and balance" to ensure products selected meet the original needs articulated by clients and stakeholders.

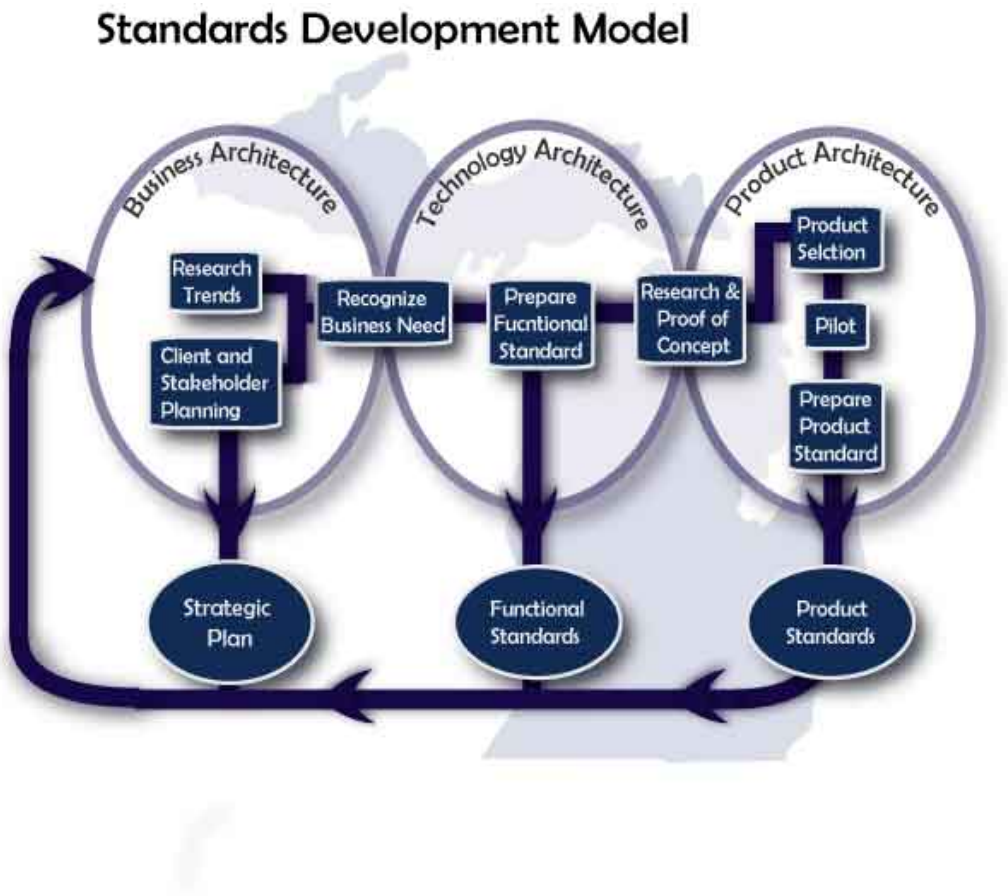


Figure 5: Standards Development Model

As detailed on the previous page the standards process follows the same basic steps as the Technical Architecture model. The emphasis within the process is on consistent and accurate business requirements followed by subsequent reviews and validations of decisions prior to a statewide commitment to any particular product or technical approach.

The process begins with information being gathered through everyday interactions such as the strategic planning process and the client relationship model. This interaction results in the *recognition of a business need.*

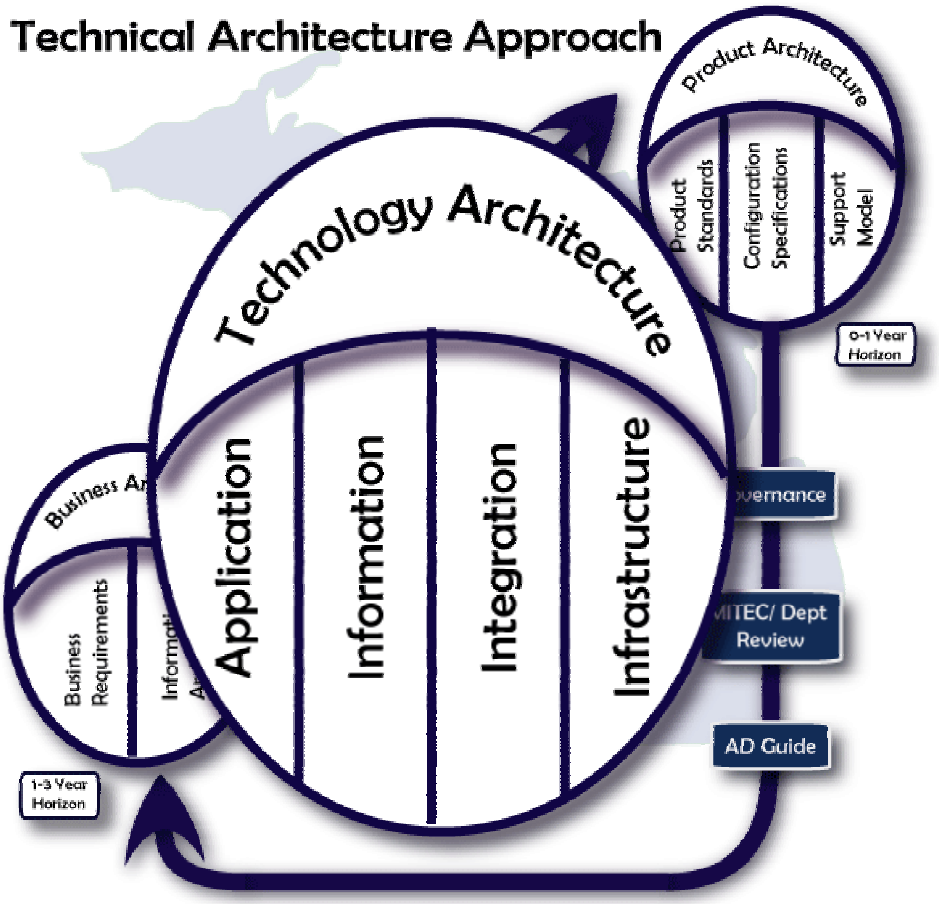
Once a business need is recognized, the standards development team *prepares the functional standard.* This process consists of requirements gathering sessions involving a cross-functional team of staff from client departments, interested parties and the Office of Strategic Policy staff. The duration, size and scope of the functional standards effort depend entirely upon the standard being reviewed or created. More effort is required for broad "enterprise-wide" standards that will affect multiple departments and stakeholders while smaller ad-hoc groups will address standards that are limited in their impact.

Once a functional standard is determined and agreed upon *research and proof of concepts* will be performed to move the process from a purely academic exercise into a sustainable "real-world" solution. The functional standard will allow the state to focus its R&D efforts on the key criteria of a successful technical solution. During the proof of concept the functional standards will be questioned for their return on investment potential, viability given the capabilities of available solutions and migration challenges faced by particular departments.

Information gathered during this phase (along with functional standards) will be used and included during a product selection and procurement phase. Once the solution is available to the state, a formal pilot of the technology will be conducted. This pilot will identify the optimal configuration, engineering issues and support models of the technology (in addition to any other associated "best practices"). These items will be documented and become part of the product standard for that given technology and its use.

The entire process is meant to be iterative and responsive to the rapidly changing technical environment. In addition to annual review of certain standards, as identified by the lifecycle process, communication of trends and impacts will occur periodically throughout the year, one key vehicle that will be utilized is the MITEC Standards Sub Committee.

A Common Language: Technology Architecture Defined



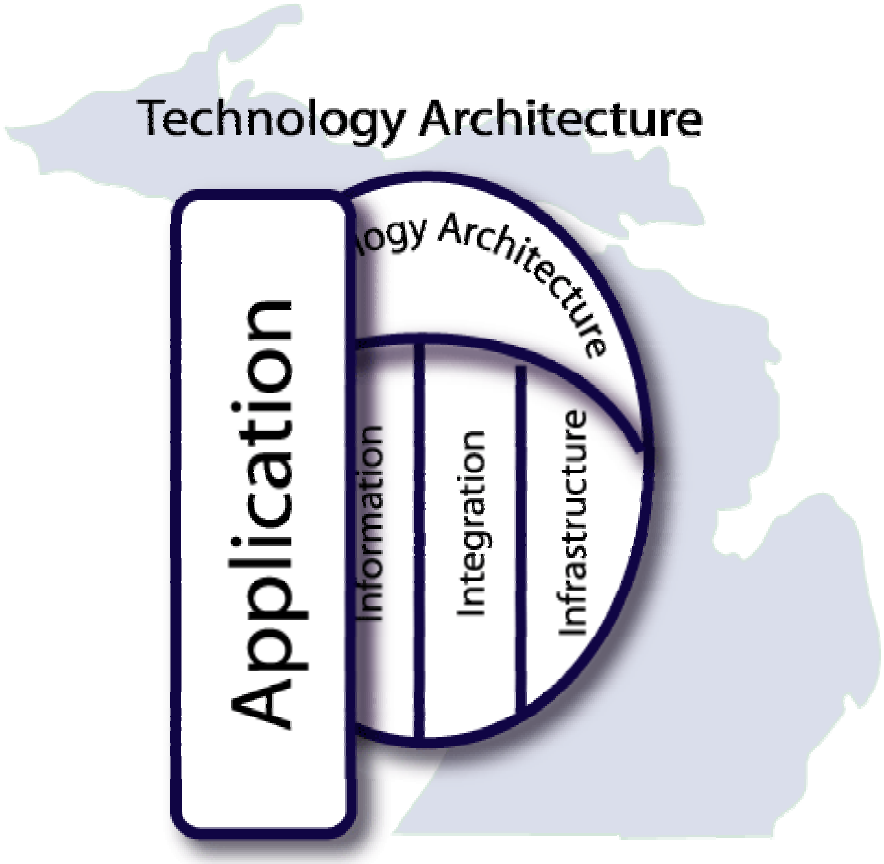
The emphasis of this document is to fully define the technology architecture framework for the State of Michigan. It can be broken down into four (4) Domains:

- Application
- Infrastructure
- Integration
- Information

Each of these domains will be detailed into more definitive disciplines, providing the basis for the selection of products and standards.

The following domain descriptions have been used to categorize the associated disciplines into logical groups that collectively support the adoption and implementation of the architecture.

Application Domain



The Application Domain defines the application environment. The application contains the business rules and requirements that provide the ability to obtain and present data to another application or a client hardware device. The technology that this domain will require will be the tools needed to design, develop, model and test application programs. Another technology that will be required by this domain is to provide oversight, process control and process management for application development, maintenance and operation. Included in this domain is the responsibility to ensure that data is presented to the client device in a consistent manner across applications.

Application Domain Detailed Breakdown

Discipline	Technology Area(s)	Specific Technology Category
Business Logic – Defines the software, protocol or method in which business rules are enforced within applications.	**Platform Independent** – Consists of all software languages that are able to execute and run on any type of operating system or platform. **Platform Dependent** – Consists of the programming languages and methods for developing software on a specific operating system or platform.	
Software Engineering – Covers not only the technical aspects of building software systems, but also management issues, such as testing, modeling and versioning.	**Modeling** – The process of representing entities, data, business logic, and capabilities for aiding in software engineering.	**Unified Modeling Language (UML)** – A general-purpose notational language for specifying and visualizing complex software, especially large, object-oriented projects. **Case Management** – Computer Aided Software Engineering (CASE) software that provides a development environment for programming teams. CASE systems offer tools to automate manage and simplify the development process.
	Test Management – The consolidation of all testing and results. Test Management activities include test planning, design (test cases), execution, reporting, code coverage and Proof of concept.	**Functional testing** – This type of test focuses on any requirements that can be traced directly to use cases (or business functions), business rules, and design. **Business Cycle Testing** – Refers to the emulation of activities performed over a period of time that is relevant to the application under test. **Performance Profiling** – Refers to the performance test that measures and evaluates response times and transaction rates. **Load/Stress/Volume Testing** – Refers to tests that measure and evaluate how a system performs and functions under varying workloads, large amounts of data and/or resource utilization. **Security and Access Control Testing** – Focuses on the technical, administrative and physical security controls that have been designed into the system architecture in order to

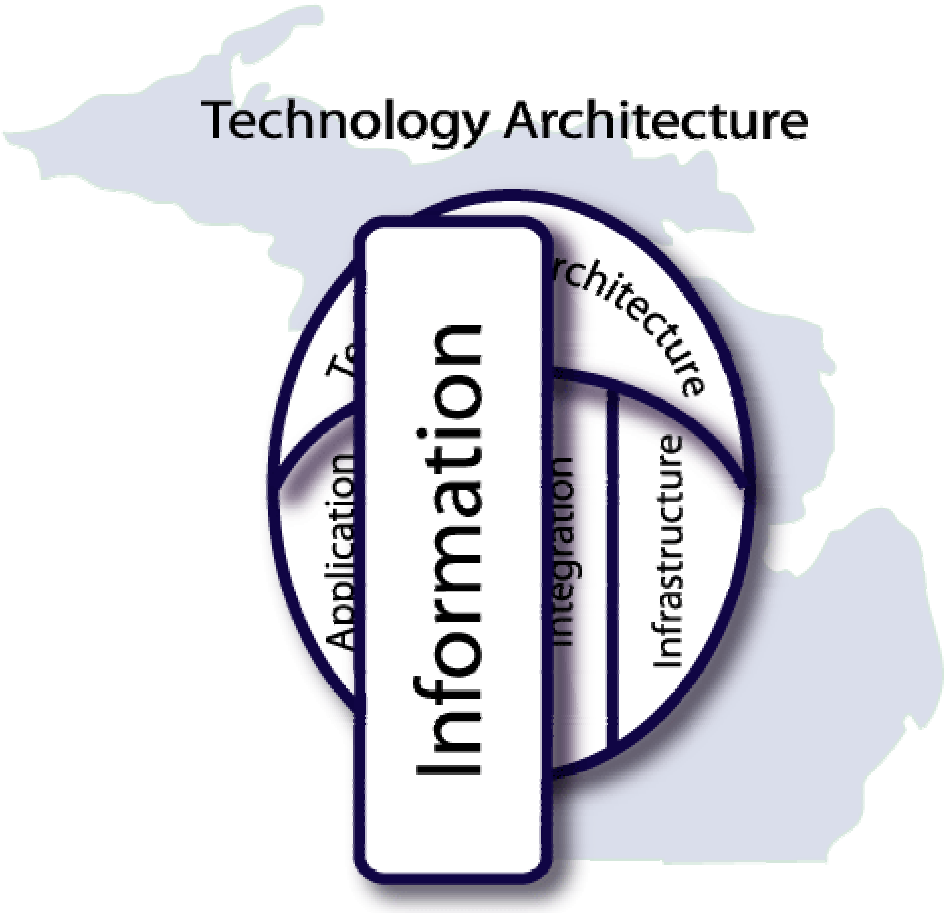
Discipline	Technology Area(s)	Specific Technology Category
		provide confidentiality, integrity and availability. **Reliability Testing** – Refers to the validation that failover methods are invoked properly and the system recovers properly. **Installation Testing** – Refers to the verification that the software installation process works properly in different environments and among varying conditions.
	Software Configuration Management – Applicable to all aspects of software development from design to delivery specifically focused on the control of all work products and artifacts generated during the development process. Several solutions on the market provide the integration of the software configuration management functions.	**Version Management** – Refers to tracking and controlling versions of files. Versions management includes capabilities such as labeling, branching, merging, version content comparisons, and security and permission management across version-controlled products. **Defect Management** – Refers to the identification, assignment, and management of discovered defects within an application, product or solution. Defect tracking tools provide searchable defect data to identify urgent and related defects or bugs. The architecture should be built to facilitate the pushing of software patches across the enterprise. **Issue Management** – refers to the management of business, technical, and infrastructure issue throughout the entire lifecycle of a product. **Task Management** – Requirements, testing, and issue assignments are transformed into prioritized tasks. Task Management tools provide automation features for managing, delivering, assigning, reminding and collaborating task management and execution. **Change Management** – Refers to the management of application code and content changes across the software development lifecycles.

Discipline	Technology Area(s)	Specific Technology Category
		Deployment Management – Refers to the capacity of software delivery to remote networked desktops, servers, and mobile devices across an enterprise. Deployment automation tolls provide centralized and accelerated delivery of applications to users via push technologies, eliminating the need for manual installation and configuration. **Requirements Management and Traceability** – Consists of information discovery, capture, storage and dissemination. Requirements management reduces software development costs and associated risks through documenting, measuring, and analyzing deviations to project requirements. Traceability refers to tracking requirements artifacts to their source and changes in requirements to include the impact analysis of the change. Requirements traceability is an integral component in quality software implementation and the management of document succession.
Integrated Development Environment (IDE) – This consists of the hardware, software and supporting services that facilitate the development of software applications and systems.	**Static Display** – Static Display consists of the software protocols that are used to create a pre-defined, unchanging graphical interface between the user and the software.	**Hyper Text Markup Language (HTML)** – The language used to create Web documents and subset of the Standard Generalized Markup Language (SGML).
	Dynamic / Server-side Display – Consists of the software that is used to create graphical user interfaces with the ability to change while the program is running.	
	Content Rendering – Defines the software and protocols used to transform data for presentation in a graphical user interface.	**Dynamic HTML (DHMTL)** – A collective term for a combination of new Hypertext Markup Language (HTML) tags and options, style sheets and programming that will allow Web pages that are more animated and more responsive to the user interaction than previous versions of HTML.

Discipline	Technology Area(s)	Specific Technology Category
		eXtensible HTML (XHTML) – The W3C's recommendation for the next generation of HTML leveraging XML. **Cascading Style Sheets (CSS)** – A style sheet format for HTML documents endorsed by the World Wide Web Consortium. CSS1 (version 1) provides hundreds of layout settings that can be applied to all subsequent HTML pages that are downloaded.
	Wireless/ Mobile/Voice – Consists of the software and protocols used for wireless and voice-enabled presentation devices.	**Wireless Markup Language (WML)** – An XML-based protocol designed for wireless devices. **XHTML mobile Profile (XHMTLMP)** – Designed for resource-constrained Web clients that do not support the full set of HTML features, such as mobile phones, PDA's, pagers and set-top boxes. It extends XHTML Basic with modules, elements and attributes to provide richer authoring language. XHTML replaces the Wireless Markup Language (WML). **Voice XML (VXML)** - VXML is an XML vocabulary for specifying the IVR (Integrated Voice Response) Systems
Access Channels – Defines the interface between application and its users, whether it is a browser, personal digital assistant or other medium.	**Web Browser** – Define the program that serves as your front end to the World Wide Web on the Internet. In order to view a site, its address (URL) is entered into the browser's locations field.	
	Wireless/PDA – Define the technologies the use transmission via the airwaves. Personal Digital Assistant (PDA) is a handheld computer that servers as an organizer for personal information. It generally includes at least a name and address database, to-do list and note taker.	
	Collaborative Communication – Define the forms of electronic exchange of messages, documents, or other information. Electronic communication provides document control, imaging, and work flow	**Electronic Mail** –E-mail (Electronic Mail) is the exchange of computer-generated and stored messages by telecommunications. An E-mail can be created manually via messaging applications or dynamically,

Discipline	Technology Area(s)	Specific Technology Category
	efficiencies.	programmatically such as automated response systems. **Kiosk** – A kiosk is a small physical structure (often including a computer and a display screen) that displays information for people walking by. Kiosks are common in public buildings. Kiosks are also used at trade shows and professional conferences.
	Other Electronic Channels – Define the other various mediums of information exchange and interface between a user and an application.	**System to System** – Involves at least two computers that exchange data or interact with each other independent of human intervention or participation. **Web Services** – Web services (sometimes called application services) are services (usually including some combination of programming and data, but possibly including human resources as well) that are made available from a business's web server for Web users or other Web-connected programs. **Uniform Resource Locator (URL)** – URL is the global address of documents and other resources on the World Wide Web. The first part of the address indicates what protocol to use (i.e. http://), and the second part specifies the IP address or the domain name where the resource is located (i.e. www.michigan.gov).

Information Domain



The information domain determines how data is stored and accessed. The most valuable asset of an Information Technology organization is the data that it maintains and stores. The ability to ensure that the data consigned to an IT organization is safe, private and easily retrievable is essential to the continued client support of that organization. The database design and functionality along with the hardware provided determines how data can be stored, retrieved and still maintain the security and privacy required by the business goals and principles. In addition to the simple store and retrieval capabilities the ability to backup, restore and provide for disaster recovery will also impact the ability to ensure save data.

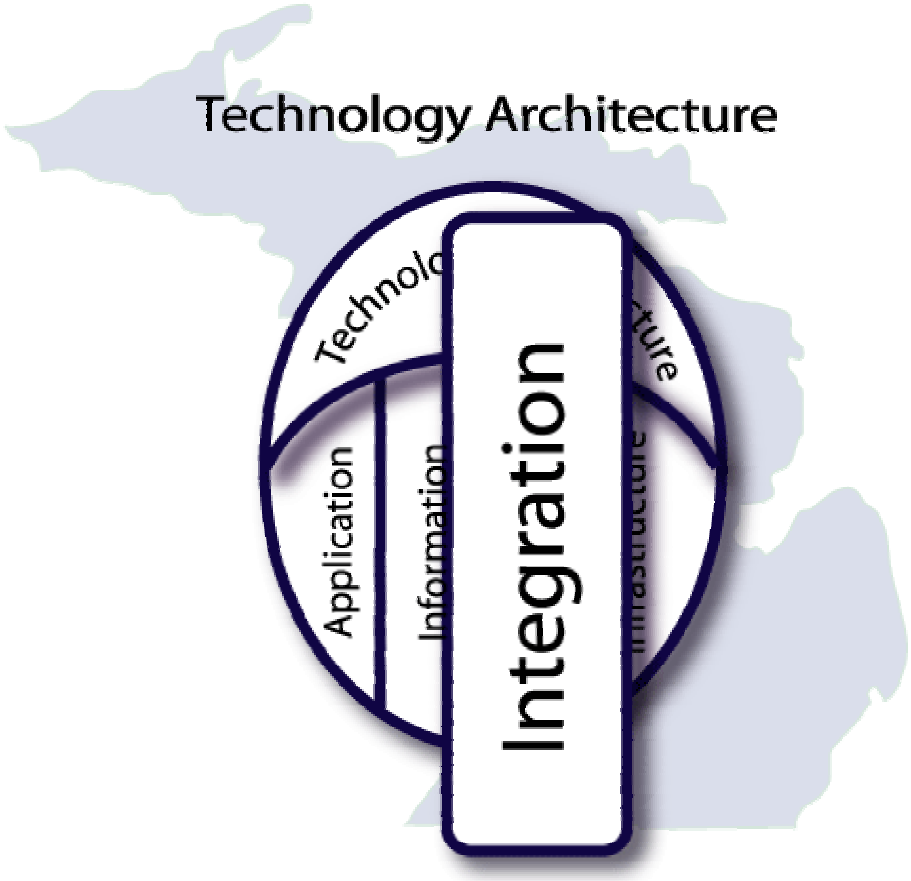
Another area that requires the safe handling of data is the ability to log important activities that occur during the operation of the enterprise. The requirement to track activity and identify the who, what, where, and when of data access for critical applications is mandatory. Auditing for both disaster recovery purposes as well as investigations of misuse will also required that logged activity be archived for a predetermined time.

Information Domain Detailed Breakdown

Discipline	Technology Area(s)	Specific Technology Category
Database / Storage - Refers to a collection of programs that enables storage, modification, and extraction of information from a database, and various techniques and devices for storing large amounts of data.	**Database** - Refers to a collection of information organized in such a way that a computer program can quickly select desired pieces of data. A database management system (DBMS) is a software application providing management, administration, performance, and analysis tools for databases.	
	Storage - Storage devices are designed to provide shared storage access across a network. These devices provide extended storage capabilities to the network with reduced costs compared to traditional file servers or local PC storage.	**Network-Attached Storage (NAS)** - A NAS device is a server that is dedicated to nothing more than file sharing. **Storage Area Network (SAN)** - A SAN is a high-speed sub-network of shared storage devices. A storage device is a machine that contains nothing but a disk or disks for storing data.
	Data Warehouses	
Data Management - The management of all data/information in an organization. It includes data administration, the standards for defining data and the way in which people perceive and use it.	**Database Connectivity** - Defines the protocol or method in which an application connects to a data store or database.	**DB2 Connector** - An IBM connectivity API to access DB2 sources.
	Reporting and Analysis - Consist of the tools, languages and protocols used to extract data from a data store and process it into useful information.	**eXtensible Business Reporting Language (XBRL)** - Extensible Business Reporting Language (XBRL is an open specification which uses XML-based data tags to describe financial statements for both public and private companies. **Online Analytical Processing (OLAP)** - Decision support software that allows the user to quickly analyze information that has been summarized into multidimensional views and hierarchies. **XML for Analysis -** XML for Analysis uses the Simple Object Access Protocol (SOAP) to let Web browser-based programs access back-end data sources for data analysis. The specification allows companies to build online analytical processing (OLAP) and data mining

Discipline	Technology Area(s)	Specific Technology Category
		applications that work over the Web.

Integration Domain



Defines the discovery, interaction and communication technologies joining disparate systems and information providers. Component-based architectures leverage and incorporate service interface and integration specifications to provide interoperability and scalability.

The integration domain describes the technology required for security access, Web services capabilities, interoperability and data integration. Technology for integration explains how interoperability between applications or database management software functions. The transformation technology guarantees a message received from a client or end-user is delivered to the desired recipient and provides translation for messages including their formats when required by communication between different applications or database management software. Security technology ensures that access to all applications and data is controlled and granted only to pre-authorized users. Security technology is also responsible for detecting unauthorized access or attempts to dispute operations by viruses or interruption of service.

Web service technology defined the portal requirements that will provide access to applications and data within the enterprise. In addition the presentation of data and information to the client device in a uniform and consistent manner will be a function of the Web services.

Integration Domain Detailed Breakdown

Discipline	Technology Area(s)	Specific Technology Category
Security - Security defines the methods of protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction in order to provide integrity, confidentiality and availability. Biometrics, two-factor identification, encryption, and technologies based on the NIST FIPS-140 standards are evolving areas of focus.	**Certificate / Digital Signature** - Software used by a certification authority (CA) to issue digital certificates and secure access to information. The evolution of Public Key Infrastructure (PKI) is based on the verification and authentication of the parties involved in information exchange.	**Digital Certificate Authentication** - Authentication implementation for controlling access to network and internet resources through managing user identification. An electronic document, digital certificate, is issued and used to prove identity and public key ownership over the network or internet.
	Secure Socket Layer - An open, non-proprietary protocol for securing data communications across computer networks. SSL is sandwiched between the application protocol (such as HTTP, Telnet, FTP, and NNTP) and the connection protocol (such as TCP/IP, UDP). SSL provides server authentication, message integrity, data encryption, and optional client authentication for TCP/IP connections.	
	Supporting Security Services - These consist of the different protocols and components to be used in addition to certificates and digital signatures.	**Secure Multipurpose Internet Mail Extensions (S/MIME)** - Provides a consistent way to send and receive secure MIME data. Based on the Internet MIME standard, S/MIME provides cryptographic security services for electronic messaging applications: authentication, message integrity and non-repudiation of origin (using digital signatures) and data confidentiality (using encryption). S/MIME is not restricted to mail; it can be used with any transport mechanism that transports MIME data, such as HTTP. **Transport Layer Security (TLS)** - Standard for the next generation SSL. TLS provides communications privacy over the Internet. The protocol allows client/server applications to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery.

Discipline	Technology Area(s)	Specific Technology Category
		Web Services Security (WS-Security) - Describes enhancements to SOAP messaging to provide message integrity, message confidentiality, and single message authentication. These mechanisms can be used to accommodate a wide variety of security models and encryption technologies including X.509, Kerberos, and SAML. **Security Assertion Markup Language (SAML)** - An XML-based framework for exchanging security information expressed in the form of assertions about subjects, where a subject is an entity (either human or computer) that has an identity in some security domain. SAML is expected to play a key role in the Federal-wide E-authentication initiative, and is supported by both the Liberty Alliance and WS-Security. **Simple Key Management Protocol (SKIP)** - A protocol developed by Sun Microsystems to handle key management across IP networks and VPNs. **Secure Shell (SSH)** - A strong method of performing client authentication. Because it supports authentication, compression, confidentiality and integrity, SSH is used frequently on the Internet. SSH has two important components, RSA certificate exchange for authentication and Triple DES for session encryption.
Interface - Interface defines the capabilities of communicating, transporting and exchanging information through a common dialog or method. Delivery Channels provide the information to reach the intended destination, whereas Interfaces allow the interaction to occur based on a predetermined framework.	**Service Discovery** - Defines the method in which applications, systems or web services are registered and discovered.	**Universal Description Discovery and Integration (UDDI)** - UDDI provides a searchable registry of XML Web Services and their associated URLs and WSDL pages.
	Service Description / Interface - Defines the method for publishing the way in which web services or	**Web Service description Language (WSDL)** - WSDL is an XML based Interface Description Language for

Discipline	Technology Area(s)	Specific Technology Category
	applications can be used.	describing XML Web Services and how to use them. **Application Program Interface (API) Protocol** - A language and message format used by an application program to communicate with the operating system or some other control program such as a database management system (DBMS) or communications protocol. APIs are implemented by writing function calls in the program, which provide the linkage to the required subroutine for execution. Thus, an API implies that some program module is available in the computer to perform the operation or that it must be linked into the existing program to perform the tasks.
Interoperability - Interoperability defines the capabilities of discovering and sharing data and services across disparate systems and vendors.	**Data Format / Classification** - Defines the structure of a file. There are hundreds of formats, and every application has many different variations (database, word processing, graphics, executable program, etc.). Each format defines its own layout of the data. The file format for text is the simplest.	**eXtensible Markup Language (XML)** - XML has emerged as the standard format for web data, and is beginning to be used as a common data format at all levels of the architecture. Many specialized vocabularies of XML are being developed to support specific Government and Industry functions. **XML Linking Language (XLINK)** - A language used to modify XML documents to include links, similar to hyperlinks, between resources. XLINK provides richer XML content through advanced linking integration with information resources. **Namespaces** - Namespaces are qualified references to URI (Uniform Resource Identifier) resources within XML documents. **Electronic Data Interchange (EDI)** - Defines the structure for transferring data between enterprises. EDI is used mainly used for purchase-related information. ANSI X.12 refers to the approved EDI standards. **Data Types / Validation** - Refers to specifications used in identifying and affirming common structures and processing rules. This technique is referenced and abstracted from the

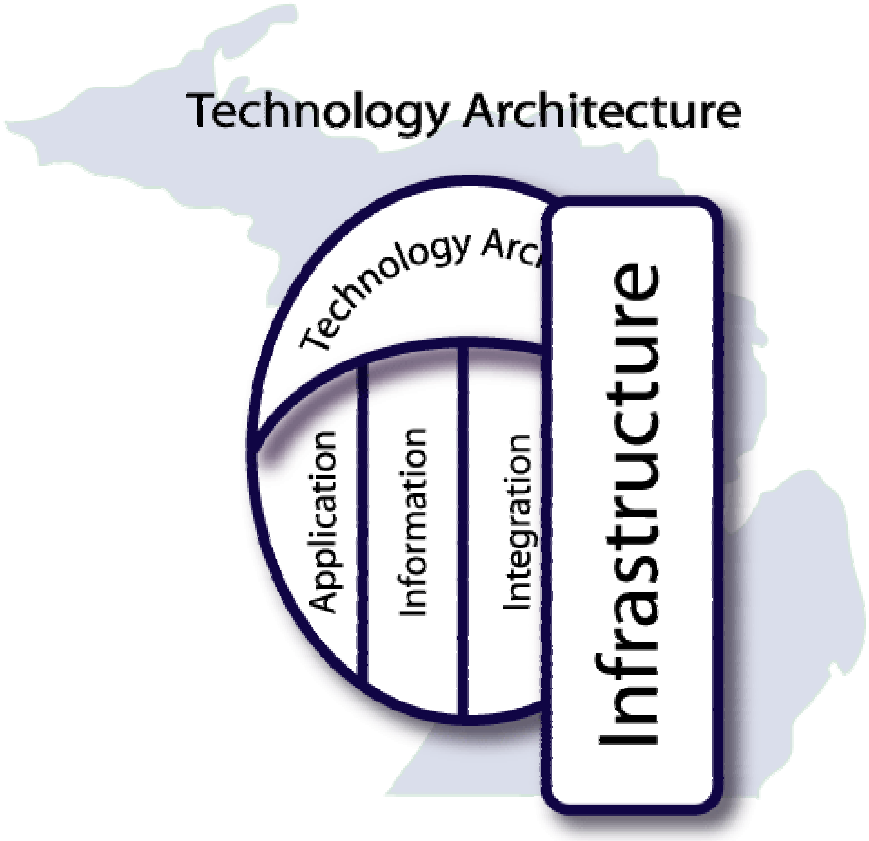
Discipline	Technology Area(s)	Specific Technology Category
		content document or source data. **Document Type Definition (DTD)** - DTD is used to restrict and maintain the conformance of an XML, HTML, or SGML document. The DTD provides definitions for all tags and attributes within the document and the rules for their usage. Alterations to the document are validated with the referenced DTD. **XML Schema** - XML Schemas define the structure, content, rules and vocabulary of an XML document. XML Schemas are useful in automation through embedding processing rules.
	Data Transformation - Data Transformation consists of the protocols and languages that change the presentation of data within a graphical user interface or application.	**eXtensible Style-sheet Language Transform (XSLT)** - Transforms XML document from one schema into another. Used for data transformation between systems using different XML schema, or mapping XML to different output devices.
Data Interchange - Define the methods in which data is transferred and represented in and between software applications.	**Data Exchange** - Data Exchange is concerned with the sending of data over a communications network and the definition of data communicated from one application to another. Data Exchange provides the communications common denominator between disparate systems.	**XMI** - Enables easy interchange of metadata between modeling tools (based on the OMG UML) and metadata repositories (OMG MOF based) in distributed heterogeneous environments. XMI integrates three key industry standards: XML, UML, and MOF. The integration of these three standards into XMI marries the best of OMG and W3C metadata and modeling technologies, allowing developers of distributed systems to share object models and other metadata over the Internet. **Xquery** - A language used for processing and evaluating XML data. The Xquery language provides results of expressions allowing the use of evaluations to the implementation of XQuery. **Simple Object Access Protocol (SOAP**) - SOAP provides HTTP/XML based remote procedure call capabilities for XML Web Services. **Electronic Business using XML (ebXML)** - A modular suite of

Discipline	Technology Area(s)	Specific Technology Category
		specifications that enables enterprises to conduct business over the Internet: exchanging business messages, conducting trading relationships, communicating data in common terms and defining and registering business processes. **Resource Description Framework (RDF)** - RDF provides a lightweight ontology system to support the exchange of knowledge on the Web. It integrates a variety of web-based metadata activities including sitemaps, content ratings, stream channel definitions, search engine data collection (web crawling), digital library collections, and distributed authoring, using XML as interchange syntax. RDF is the foundation for the Semantic Web envisioned by Tim Berners-Lee - an extension of the current web in which information is given well defined meaning, better enabling computers and people to work in cooperation. **Web Services User Interface (WSUI)** - WSUI uses a simple schema for describing a WSUI "component" that can be used in a portal to call backend SOAP and XML services. WSUI uses XSLT style-sheets to construct user-facing views to enable users to interact with the services.
Integration - Integration defines the software services enabling elements of distributed business applications to interoperate. These elements can share function, content, and communications across heterogeneous computing environments. In particular, service integration offers a set of architecture services such as platform and service location transparency, transaction management, basic messaging between two points, and guaranteed message delivery.	**Middleware** - Middleware increases the flexibility, interoperability, and portability of existing infrastructure by linking or "gluing" two otherwise separate applications.	**Remote Procedure Call (RPC)** - RPC is a protocol allowing a program on a client computer to invoke a program on a server computer. **Message-Oriented Middleware (MOM** - Software solution providing APIs, queue management, message routing, automatic fail-over, and workload balancing. Message-Oriented Middleware (MOM) is software residing in both sides of the client/server architecture providing support for asynchronous calls, or messages, between applications. Message queues are used to track and store requests waiting for execution by the source application. Messaging allows otherwise complex programming and networking details to

Discipline	Technology Area(s)	Specific Technology Category
		be abstracted from the developer. **Transaction Monitor** - Software providing synchronous messaging and queuing along with other transaction management services designed to support the efficient processing of high volumes of transactions. Core services include load balancing, rollback/commit, and recovery. Transaction Processing provides cost-effective scalability to applications and database systems by managing and throttling transactions on behalf of the database system. **Object Request Broker (ORB): Common Object Request Broker Architecture (CORBA)** – An architecture that enables objects to communicate with one another regardless of what programming language they were written in or what operating system they're running on. Object Request Broker (ORB) is a technology enabling distributed objects to communicate and exchange data with remote objects. ORB encapsulates the locality and implementation of the objects, allowing users to develop applications that leverage components by accessing the components interface. **Object Request Broker (ORB): Compound Object Model (COM) - Component Object Model (COM)** – A software architecture to design and build component-based applications. COM object capabilities are accessible from exposed interfaces. Object Request Broker (ORB) is a technology enabling distributed objects to communicate and exchange data with remote objects. ORB encapsulates the locality and implementation of the objects, allowing users to develop applications that leverage components by accessing the components interface. **Object Request Broker (ORB): Distribute Compound Object Model (DCOM)** – An extension of the Component Object Model (COM) that allows COM components to

Discipline	Technology Area(s)	Specific Technology Category
		communicate across network boundaries. Traditional COM components can only perform inter-process communication across process boundaries on the same machine. Object Request Broker (ORB) is a technology enabling distributed objects to communicate and exchange data with remote objects. ORB encapsulates the locality and implementation of the objects, allowing users to develop applications that leverage components by accessing the components interface. **Object Request Broker (ORB): Compound Object Model + (COM+)** - COM+ is an extension of the COM that provides a runtime and services that are readily used from any programming language or tool, and enables extensive interoperability between components regardless of how they were implemented. Object Request Broker (ORB) is a technology enabling distributed objects to communicate and exchange data with remote objects. ORB encapsulates the locality and implementation of the objects, allowing users to develop applications that leverage components by accessing the components interface.
	Enterprise Application Integration - Refers to the processes and tools specializing in updating and consolidating applications and data within an enterprise. EAI focuses on leveraging existing legacy applications and data sources so that enterprises can add and migrate to current technologies.	**Business Process Management** - This process is responsible for the definition and management of cross-application business processes across the enterprise and/or between enterprises. **Application Connectivity** - This process provides reusable, non-invasive connectivity with packaged software. Uni- or bidirectional adapters provide this connectivity. **Transformation and Formatting** - This process is responsible for the conversion of data, message content, information structure, and syntax to reconcile differences in data amongst multiple systems and data sources.

Infrastructure Domain



The Infrastructure Domain describes the network design and functionality, the configuration and capabilities of the data center and the operation of the platforms required to support the applications, data storage and the communications required by the entire architecture. The infrastructure will be asked to provide the standard services to support an application environment that is diverse and demanding in its operation and reliability. The infrastructure will be required to support critical applications that must be operational 24 x 7 and must have full redundancy and disaster recovery available to applications that will only be required to be available during normal working hours with minimal recovery capabilities.

Infrastructure Domain Detailed Breakdown

Discipline	Technology Area(s)	Specific Technology Category
Supporting Platform – Supporting platforms are hardware and software architectures. The term originally dealt with only hardware, and it is still used to refer to a CPU model or computer family.	**Wireless/ Mobile** – Radio transmissions via the airwaves. Various communications techniques are used to provide wireless transmission including infrared line of sight, cellular, microwave, satellite, packet radio and spread spectrum.	
	Platform Independent – Defines the operating systems and programming languages that are able to execute and run on any platform or operating system. A platform is the underlying hardware and software comprising a system.	
	Platform Dependent – Defines the operating system and programming languages that are able to execute and run on a specific platform or operating system. A platform is the underlying hardware and software comprising a system.	
Delivery Servers – Delivery Servers are front-end platforms that provide information to a requesting application. It includes the hardware, operating system, server software, and networking protocols.	**Web Servers** – A computer that provides World Wide Web services on the Internet. It includes the hardware and operating system, Web server software, TCP/IP protocols and the Web site content (Web Pages). If the Web server is used internally and not by the public, it may be known as an "intranet server".	
	Media Servers – Provide optimal management of media-based files such as audio and video streams and digital images.	
	Application Servers – In a three-tired environment, a separate computer (application server) performs the business logic, although some part may still be handled by the user's machine.	
	Portal Servers – Portals represent focus points for interaction, providing integration and single-source corporate information.	

Discipline	Technology Area(s)	Specific Technology Category
Hardware/Infrastructure – Defines the physical devices, facilities and standard technologies that provide the computing and networking within and between enterprises.	**Servers / Computers** – Refers to the various types of programmable machines which are capable of responding to sets of instructions and executing programs.	**Enterprise Servers** – A computer or device on a network that manages network resources and shares application for multiple users. **Mainframes** – A very large computer capable of supporting hundreds or even thousands of users simultaneously.
	Embedded Technology Device – Refers to the various devices and parts that make up a server or computer as well as devices that perform specific functionalities outside of a server or computer.	
	Peripherals – Computer device that are not part of the essential computer. Peripheral devices can be external or internal.	
	Wide Area Network – A data network typically extending a LAN outside a building or beyond a campus. Typically created by using bridges or routers to connect geographically separate LANs. WANs include commercial or educational dial-up networks such as CompuServe, InterNet and BITNET.	
	Local Area Network – A network that interconnects devices over a geographically small area, typically in one building or part of a building. The most popular LAN type is Ethernet. LANs allow the sharing of resources and the exchange of both video and data.	
	Network Devices/ Standards – A group of stations (computers, telephones, or other devices) connected by communication facilities for exchanging information. Connection can be permanent, via cable, or temporary, through telephone or other communication links. The transmission medium can be physical (i.e. fiber optic cable) or wireless (i.e. satellite).	
	Video Conferencing - Communication across long distances with video and audio contact that may	**Bridge -** A bridge connects three or more conference sites so that they can simultaneously pass data, voice, or

Discipline	Technology Area(s)	Specific Technology Category
	also include graphics and data exchange. Digital video transmission systems typically consist of camera, codec (coder-decoder), network access equipment, network, and audio system.	video. Videoconferencing bridges are often called MCUs (multipoint conferencing units). **CODEC -** A video codec converts analog video signals from a video camera to digital signals for transmission over digital circuits, and then converts the digital signals back to analog signals for display.
Delivery Channels - Delivery Channels define the level of access to applications and systems based upon the type of network used to deliver them.	**Internet** - The Internet is a worldwide system of computer networks in which users at any one computer can, if they have permission, get information from any other computer.	
	Intranet - An Intranet is a private network that is contained within an enterprise. It may consist of many inter-linked local area networks and is used to share company information and resources among employees.	
	Extranet - An Extranet is a private network that uses the Internet protocol and the public telecommunication system to securely share part of a business's information or operations with suppliers, vendors, partners, customers, or other businesses. An extranet can be viewed as part of a company's intranet that is extended to users outside the company.	
	Peer to Peer (P2P) - Peer-to-Peer is a class of applications that operate outside the DNS system and have significant or total autonomy from central servers that take advantage of resources available on the Internet.	
	Virtual Private Network - A private data network that makes use of the public telecommunication infrastructure, maintaining privacy through the use of a tunneling protocol and security procedures.	
Service Transport - Service Transport defines the end-to-end management of the communications session to include the access and delivery protocols.	**Supporting Network Services** - Consist of the protocols that define the format and structure of data and information that is either accessed from a directory or exchanged through communications.	**Internet Message Access Protocol / Post Office Protocol (IMAP/POP3)** – IMAP allows a client to access and manipulate electronic mail messages on a server. IMAP permits manipulation of remote message folders, called

Discipline	Technology Area(s)	Specific Technology Category
		"mailboxes", in a way that is functionally equivalent to local mailboxes. IMAP also provides the capability for an offline client to resynchronize with the server. POP3 is the most commonly used protocol for retrieving e-mail from a mail host. (Refers to RFC2060) **Multipurpose Internet Mail Extensions (MIME)** – MIME extends the format of Internet mail to allow non-US- American Standard Code for Information Interchange (ASCII) textual messages, non-textual messages, multi-part message bodies, and non-US-ASCII information in message headers. MIME support allows compliant email clients and servers to accurately communicate embedded information to internal and external users. (Refers to RFC 2045) **Simple Mail Transfer Protocol (SMTP)** – SMTP facilitates transfer of electronic-mail messages. It specifies how two systems are to interact, and the messages format used to control the transfer of electronic mail. (Refers to RFC821) **Extended Simple Mail Transfer Protocol (ESMTP)** - ESMTP allows new service extensions to SMTP to be defined and registered with Internet Assigned Numbers Authority (IANA). (Refers to RFC1869) **Simple Network Management Protocol (SNMP)** - Simple Network Management Protocol (SNMP) is the protocol governing network management and the monitoring of network devices and their functions. It is not necessarily limited to TCP/IP networks. **Lightweight Directory Access Protocol (LDAP)** - LDAP is a subset of X.500 designed to run directly over the TCP/IP stack. LDAP is, like X.500, both an information model and a protocol for querying and manipulating it. LDAPv3 is an update

Discipline	Technology Area(s)	Specific Technology Category
		developed in the IETF (Internet Engineering Task Force), which address the limitations found during deployment of the previous version of LDAP. (Refers to LDAP V3, RFC 1779) **Directory Service (X.500)** – This is a network service that discovers and identifies resources on a network and makes them accessible to users and applications. The resources include users, e-mail addresses, computers, mapped drives, shared folders, and peripherals such as printers and PDA docking stations. Users and computers access these resources without the needing to know how or where the resources are connected. **Dynamic Host Configuration Protocol (DHCP)** – A protocol for assigning dynamic IP addresses to devices on a network. A device can receive a different IP address for every connection. Dynamic addressing provides reduced network administration over deploying and connecting user and peripheral devices. **Domain Name System (DNS)** – A protocol used for translating domain names (i.e. www.feapmo.gov) to their respective IP addresses. DNS is collectively a network of devices that store query results. As one DNS server or device cannot provide the translated IP address, it queries other DNS devices. This process is invisible to the user. **Border Gateway Protocol (BGP)** – Refers to a routing protocol used to exchange routing information between routers on a network, enabling more efficient routing of data. BGP is part of RFC 1771.
	Transport Protocols- These consist of the protocols that define the format and structure of data and information that is either accessed from a directory or exchanged through communications.	**Transport Control Protocol (TCP)** - TCP provides transport functions, which ensures that the total amount of bytes sent is received correctly at the destination.

Discipline	Technology Area(s)	Specific Technology Category
		Internet Protocol (IP) - This is the protocol of the Internet and has become the global standard for communications. IP accepts packets from TCP, adds its own header and delivers a "datagram" to the data link layer protocol. It may also break the packet into fragments to support the maximum transmission unit (MTU) of the network. **Hyper Text Transfer Protocol (HTTP)** - The communications protocol used to connect to servers on the World Wide Web. Its primary function is to establish a connection with a web server and transmit HTML pages to the client browser. **Hyper Text Transfer Protocol Secure (HTTPS)** - The protocol for accessing a secure Web server. Using HTTPS in the URL instead of HTTP directs the message to a secure port number rather than the default Web port number of 80. The session is then managed by a security protocol. **Wireless Application Protocol (WAP)** - The Wireless Application Protocol (WAP) is an open, global specification that empowers users of digital mobile phones, pagers, personal digital assistants and other wireless devices to securely access and interact with Internet/intranet/extranet content, applications, and services. **File Transfer Protocol (FTP)** - A protocol used to transfer files over a TCP/IP network (Internet, UNIX, etc.). For example, after developing the HTML pages for a Web site on a local machine, they are typically uploaded to the Web server using FTP. **IP Security (IPSEC)** - A set of protocols used to secure IP packet exchange. Tunnel and Transport are the two (2) modes supported by IPSEC. IPSEC uses certificates and Public Keys to authenticate and validate the sender and receiver.